#23
9K
4-15-03

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of:			
Prudvi, et. al.			
Serial No.:	09/212,291	Examiner:	Tuan V. Thai
Filing Date:	December 16, 1998	Art Unit:	2186
Title:	TRANSACTION MANAGER AND CACHE FOR PROCESSING AGENT		

REPLY BRIEF

ASSISTANT COMMISSIONER FOR PATENTS
Washington, DC 20231

Sir:

Appellants respectfully submit this Reply Brief in the above-referenced appeal. The Examiner's Answer fails to address many of the points made in Appellants' Appeal Brief. It largely repeats the analysis from prior office actions. Some of Appellants' arguments go entirely unanswered. Appellants maintain their contention that the pending claims define patentable subject matter over both Sachs and Scales, and therefore request this Board to reverse all outstanding rejections.

As Appellants have explained in earlier submissions, embodiments of the present invention break a one-to-one relationship of data line length to cache line length that is found in the prior art. Conventionally, prior systems had internal caches that were constructed to store, in each cache entry (or cache *line*), an amount of data that corresponded to the maximum amount of data that could be transferred in a single bus transaction. The present invention introduces a new organization of cache memory, where each cache line is able to store several times the amount of data transferred in a single bus transaction (a single data line corresponds to the maximum amount of data that can be transferred in a single bus transaction).

Each of the independent claims includes an element that describes this important many-to-one relationship:

Claim 1 recites: "A processing agent ... comprising an internal cache having a plurality of cache entries, *each entry sized to store multiple data line lengths of data*."

Claim 11 recites: "A processing agent, comprising: an internal cache *having cache entries each sized to store multiple data lines*...."

Claim 17 recites: "A method of processing a data request ... *wherein each cache line is sized to store multiple data line lengths of data*."

Claim 23 recites: "A method of processing a data request ... *wherein each cache line is sized to store multiple data line lengths of data*."

Claim 24 recites: "A processing agent, comprising: an internal cache having a plurality of cache lines, *each cache line including: ... a plurality of cache entries, each cache entry sized to store one data line length of data*...."

The Examiner's Answer has failed to demonstrate that the claimed many-to-one relationship is shown in the cited references.

The issues remaining on appeal, following a withdrawal by the Examiner of an earlier rejection, are as follows:[1]

1. Whether claims 24 and 29 patentably distinguish over Sachs, U.S. Patent No. 4,884,197 ("Sachs") under 35 U.S.C. § 102(b);
2. Whether claims 17-21, 23 and 27-28 patentably distinguish over Scales III, et al., U.S. Patent No. 4,914,573 ("Scales") under 35 U.S.C. § 102(b); and
3. Whether claims 1-7, 11-16 and 25-26 patentably distinguish over Sachs in view of Scales under 35 U.S.C. § 103(a).

Sachs Fails to Teach Every Element of Claims 24 and 29.

The Examiner's Answer fails to respond to the substance of Appellants' arguments concerning claims 24 and 29. Instead, the Examiner argues that the important many-to-one relationship is irrelevant with respect to claim 24 and claim 29. Citing language in claim 24, the Examiner contends that claim 24 describes a cache line that can only store one data line length of data. Respectfully, the Examiner is mistaken. Claim 24 recites:

---

[1] In the Examiner's Answer, the earlier rejection of claims 17-21, 23-24 and 27-29 as being anticipated by Sachs, U.S. Patent No. 4,884,197 was withdrawn.

24. A processing agent, comprising:

an internal cache having a plurality of cache lines, *each cache line including*:

a tag portion storing address information, and

a plurality of cache entries, *each cache entry sized to store one data line length of data*;

wherein the processing agent, in response to a multiple transaction signal, posts a series of external transactions related to the address information, each of said external transactions filling one of said cache entries in the cache line.

Claim 24 describes a hierarchy of structures. The outermost structure is a cache. Within the cache is a plurality of cache lines. Within each cache line is a *plurality of cache entries*, each of which are sized to store one data line length of data. Thus, rather than describing cache lines that are able to store only one data line length of data, claim 24 describes cache lines that can store multiple data line lengths of data, and are filled by multiple bus transactions.

Each of Sachs' cache lines can hold a maximum of 128 bits of information. See, e.g., Sachs, col. 22:45-48. Additionally, Sachs' data line length – the maximum amount of data that Sachs can transfer in a single bus transaction – is 512 bits. See, e.g., Sachs, col. 10:24-34; 14:5-8. Thus, Sachs cannot possibly teach or suggest what the present invention claims: a cache line that can store *multiple* data line lengths. The anticipation rejection to claims 24 and 29 must be reversed.

Scales Fails to Teach Every Element of Claims 17-21, 23 and 27-28.

The Examiner's Answer attempts to show that Scales describes Appellants' many-to-one relationship. First, the Examiner points to the last sentence in Scales' specification, which states, "[i]f the size of the entries in a cache line is different, the size detecting criteria of the state machine 24 must be appropriately adjusted." Then, the Examiner attempts to support his position by quoting from Scales' summary of the invention, which declares "it is an object of the present invention to provide a bus master which selectively attempts to fill entire *entries* in a cache line." (retaining the Examiner's emphasis on the plural form).

Appellants acknowledge that Scales describes cache lines that contain more than one entry. Appellants further acknowledge that the size of entries in Scales' cache lines may be different. However, these features fail to anticipate claims 17-21, 23 and 27-28 because Scales does not disclose a method of posting external transactions in order to *fill an entire cache line*. Scales only fills *entries* within a cache line. Scales never fills an entire cache line.

Independent claims 17 and 23 each disclose "posting a sequence [or series] of external transactions ... to *fill a cache line...*" (emphasis added). Scales fails to teach or disclose this feature. Throughout Scales' disclosure, whenever a memory request fails to fill a particular cache *entry*, Scales issues additional memory requests that are designed to fill that cache *entry* with data that was "missing" from the initial memory request. Contrary to the Examiner's assertion, Scales contains no teaching to *fill an entire cache line* in response to a memory request. Thus, the § 102(b) rejection of claims 17-21, 23 and 27-28 must be reversed.

Sachs and Scales Together Do Not Teach or Suggest Every Element of the Claims.

Appellants' nonobvious arguments with respect to claims 1-7, 11-16 and 25-26 were advanced in their Appeal Brief submitted previously. The Examiner has not responded to these arguments. Instead, the Examiner has merely reasserted, without sufficient justification, that the earlier § 103(a) rejection is proper because: (a) Sachs and Scales are in the same field of endeavor; and (b) combining Scales with Sachs would make Sachs more flexible. This is not enough. The Examiner has failed to demonstrate how one of ordinary skill in the art would be led solely by the teachings of Sachs and Scales to combine them in a manner that achieves the present invention.

Reading between the lines, the Examiner's Answer seems to hint that one of ordinary skill in the art would be motivated to insert Scales' cache memory structure into Sachs. However, giving the Examiner every possible consideration, this combination would still fail to describe the present invention. The reason is this: Scales' cache line length is not a multiple of Sachs' data line length. Sachs' data line length is 512 bits. To reach the present invention, Scales would need a cache line that is at least 1,024 bits wide (to hold at least two data line length's of data). But Scales' cache line is composed of 4 cache entries, each of which is 32-bits wide. Thus, Scales' cache line is only 128 bits wide. As the Examiner has noted, Scales teaches that "if the size of the entries in a cache line is different, the size detecting criteria of the state machine 24 must be appropriately adjusted." Scales, col. 7:59-62. However, this state machine adjustment is not directed to producing a many-to-one ratio of data line length to cache line length. Rather, it is directed to the behavior of *state machine 24*, which is designed to continue issuing memory requests to fill bytes in a cache entry that might be left unfilled after a given operand request. Scales, col. 3:14 to col. 4:24. The Examiner's rejection under § 103(a) of claims 1-7, 11-16 and 25-26 should be reversed.

Conclusion

Appellants respectfully request the Board to reverse all outstanding rejections to the pending claims and to remand the case for further consideration by the Examiner.

In a paper filed separately, Appellants request an oral hearing under 37 C.F.R. § 1.194. The Examiner's Answer was dated February 10, 2003, so this Request is timely filed. The $280.00 fee required under § 1.17(d) is authorized to be charged to Kenyon & Kenyon's Deposit Account 11-0600.

Respectfully submitted,

Date: April 10, 2003

Clyde E. Findley
Reg. No. 50,724

KENYON & KENYON
1500 K Street, NW
Washington, DC 20005
Phone: (202) 220-4200
Facsimile: (202) 220-4201